

*A1*  
*Cancelled*  
commonly-owned US Patent Application, Attorney Docket No. CYPR-CD00232, Serial No. 10/033,027, filed October 22, 2001, by W. Snyder, and entitled "Programmable Microcontroller Architecture," hereby incorporated by reference.

*Please replace the paragraph beginning at page 4, line 17 with the following new paragraph:*

*A2*  
Thus, what is needed is a method or system that can help guide a user through a series of tasks in an orderly manner while facilitating movement between tasks. What is also needed is a method or system that can satisfy the above need and that can be used for the design of microcontrollers, such as microcontrollers of the design mentioned above. The present invention provides a novel solution to these needs.

*Please replace the paragraph beginning at page 8, line 9 with the following new paragraph:*

*A3*  
Figure 2A is a block diagram of an exemplary programmable single-chip system architecture used with one embodiment of the present invention.

*Please replace the paragraph beginning at page 8, line 12 with the following new paragraph:*

*A4*  
Figure 2B is a block diagram of an exemplary arrangement of programmable system blocks used with one embodiment of the present invention.

*Please replace the paragraph beginning at page 13, line 11 with the following new paragraph:*

*A5*  
*em't*  
The present invention is described in the context of a software tool, portions of which are comprised of computer-readable and computer-executable instructions which reside, for example, in computer-usable media of a computer system such as that exemplified by Figure 1. The present invention is primarily described as being used with a tool for designing, configuring, programming, compiling, building (assembling), emulating, and debugging an embedded microcontroller, in

*A5*  
*uncl* particular a class of microcontrollers that provide analog and/or digital subsystems comprising many dynamically configurable blocks. An example of this class is referred to herein as a programmable microcontroller architecture. Additional information regarding such an architecture is provided in the co-pending, commonly-owned US Patent Application, Attorney Docket No. CYPR-CD00232, Serial No. 10/033,027, filed October 22, 2001, by W. Snyder, and entitled "Programmable Microcontroller Architecture," hereby incorporated by reference.

---

*Please replace* ~~the paragraph beginning at page 14, line 4 with the following new paragraph:~~

*A6* Figure 2A is a block diagram of an integrated circuit (or microcontroller) 210 that exemplifies a microcontroller which uses a programmable architecture. In the illustrated embodiment, integrated circuit 210 includes a system bus 211, and coupled to bus 211 are synchronous random access memory (SRAM) 212 for storing volatile or temporary data during firmware execution, central processing unit (CPU) 214 for processing information and instructions, flash read-only memory (ROM) 216 for holding instructions (e.g., firmware), input/output (I/O) pins 218 providing an interface with external devices and the like, and system blocks 225. The system blocks 225 include analog blocks and digital blocks.

---

*Please replace* ~~the paragraph beginning at page 14, line 15 with the following new paragraph:~~

*A7*  
*Cm't* Referring to Figure 2B, an embodiment of system block 225 is depicted in greater detail. In this embodiment, system block 225 includes an analog functional block 230, a digital functional block 240, and a programmable interconnect 250. Analog block 230 includes, in the present embodiment, a matrix of interconnected analog blocks A1 through AN. The number N may be any number of analog blocks. Likewise, digital block 240 includes, in the present embodiment, a matrix of interconnected digital blocks D1 through DM. The number M may be any number of digital blocks. The analog blocks A1 through AN and the digital blocks D1 through DM are fundamental

*A7*  
*modd.* building blocks that may be combined in different ways to accomplish different functions. Importantly, different combinations of blocks, producing different functions, may exist at different times within the same system. For example, a set of blocks configured to perform the function of analog-to-digital conversion may sample a signal. After processing that signal in the digital domain, those same blocks (perhaps in conjunction with a few others) may be recombined in a different configuration to perform the function of digital-to-analog conversion to produce an output signal.

---

*Please replace the paragraph beginning at page 16, line 4 with the following new paragraph:*

*A8* With reference next to Figure 3, process 300 illustrates exemplary steps used by a microcontroller design tool in accordance with one embodiment of the present invention. The purpose of process 300 is to configure, program, compile, build, emulate and debug a customized microcontroller (a “target device”) based on the integrated circuit 210 and system blocks 225 of Figures 2A and 2B.

---

*Please replace the paragraph beginning at page 17, line 7 with the following new paragraph:*

*A9* In step 310, applicable “user modules” are selected. A user module, as used herein, is a preconfigured function that may be based on more than one system block. A user module, once placed and programmed, will work as a peripheral on the target device. At any time in process 300, user modules may be added to or removed from the target device.

---

*Please replace the paragraph beginning at page 17, line 13 with the following new paragraph:*

*A10* The selected user modules can then be “placed” or “mapped” onto the system blocks 225 of Figure 2B. Once a user module is placed, its parameters can be viewed and modified as needed. Global parameters used by all of the user modules (for example, CPU clock speed) can also be set.

---

~~Please replace~~ the paragraph beginning at page 17, line 18 with the following new paragraph:

Continuing with step 310 of Figure 3, interconnections between the selected user modules **A11** can be specified, either as each user module is placed or afterwards. The pin-out for each programmable system block can be specified, making a connection between the software configuration and the hardware of the target device.

~~Please replace~~ the paragraph beginning at page 21, line 4 with the following new paragraph:

In the present embodiment, some of the elements are always active, regardless of which task **A12** is being performed (or which window or workspace is open). As depicted in Figure 4A, the three elements A, B and C are active at all times, and the other elements are not active. For example, the microcontroller design tool described in conjunction with Figure 3 is, in one embodiment, divided into at least three subsystems: a Device Editor subsystem, an Application Editor subsystem, and a Debugger subsystem. In essence, the Device Editor subsystem implements steps 310-320 of Figure 3, the Application Editor subsystem implements steps 330-340 of Figure 3, and the Debugger subsystem implements step 360 of Figure 3. According to the present invention, an element is associated with each of these subsystems, and the element for each subsystem is always active, regardless of which subsystem is being used for the task at hand. Thus, a user can readily move between various tasks (or windows or workspaces) by selecting the appropriate active element. For example, in the microcontroller design tool, the user can move from the Debugger subsystem back to the Device Editor subsystem without having to pass through the Application Editor subsystem.

~~Please replace~~ the paragraph beginning at page 21, line 21 with the following new paragraph:

Another feature of the present invention is that the graphic elements are rendered in GUI 400 **A13** in locations that correspond to the logical order in which tasks should be performed. For example, **Com. T** as described above, in the microcontroller design tool, the Device Editor subsystem implements steps

310-320 of Figure 3, the Application Editor subsystem implements steps 330-340 of Figure 3, and the Debugger subsystem implements step 360 of Figure 3. In accordance with the present invention, element A is associated with the Device Editor subsystem, element B with the Application Editor subsystem, and element C with the Debugger subsystem. The order of the graphic elements is used to suggest to the user the order in which the subsystems are to be accessed.

---

*A13*  
*concl.*  
Please replace the paragraph beginning at page 22, line 16 with the following new paragraph:

*A14*  
Figure 4B is described further by way of example. In the microcontroller design tool, a user can select a user module and place it using the Device Editor subsystem (refer to step 310 of Figure 3). To accomplish this, the user selects element A (for example) to implement the Device Editor subsystem. In response to this selection, elements for the user module selection task and for the user module placement task (e.g., elements A1 and A2) are activated. Other elements may also be activated for the other tasks that can be performed using the Device Editor subsystem.

---

*A15*  
Please replace the paragraph beginning at page 23, line 1 with the following new paragraph:

Thus, similar to that described above, elements are presented to the user in such a way so as to guide the user through the tasks in a logical order; for example, the user module selection task and the user module placement task are not activated until the user implements the Device Editor subsystem. As well, the elements are placed to suggest to the user a logical order for performing the tasks. That is, for example, the elements for the user module selection task and for the user module placement task are proximate to each other, implying a relationship between the tasks. Also, the element for the user module selection task is placed to the left of the element for the user module placement task, implying an order in which the tasks should be performed.

---

Please replace the paragraph beginning at page 25, line 7 with the following new paragraph: